20-MAD - 20 Years of Issues and Commits of Mozilla and Apache Development

Maëlick Claes
M3S, ITEE, University of Oulu, Finland
maelick.claes@oulu.fi

Mika V. Mäntylä
M3S, ITEE, University of Oulu, Finland
mika.mantyla@oulu.fi

ABSTRACT

Data of long-lived and high profile projects is valuable for research on successful software engineering in the wild. Having a dataset with different linked software repositories of such projects, enables deeper diving investigations. This paper presents 20-MAD, a dataset linking the commit and issue data of Mozilla and Apache projects. It includes over 20 years of information about 765 projects, 3.4M commits, 2.3M issues, and 17.3M issue comments, and its compressed size is over 6 GB. The data contains all the typical information about source code commits (e.g., lines added and removed, message and commit time) and issues (status, severity, votes, and summary). The issue comments have been pre-processed for natural language processing and sentiment analysis. This includes emoticons and valence and arousal scores. Linking code repository and issue tracker information, allows studying individuals in two types of repositories and provide more accurate time zone information for issue trackers as well. To our knowledge, this the largest linked dataset in size and in project lifetime that is not based on GitHub.

CCS CONCEPTS

• **Software and its engineering → Open source model**; **Software version control**; • **Computing methodologies → Natural language processing**.

ACM Reference Format:
Maëlick Claes and Mika V. Mäntylä. 2020. 20-MAD - 20 Years of Issues and Commits of Mozilla and Apache Development. In *17th International Conference on Mining Software Repositories (MSR '20), October 5–6, 2020, Seoul, Republic of Korea.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3379597.3387487

1 INTRODUCTION

Data of long-lived and high profile projects is valuable for research on successful software engineering in the wild. Having a dataset with different linked software repositories of such projects, enables deeper diving investigations. Yet, linking data from different software repositories, such as code repositories and issue trackers, for mining purpose can be challenging. During the past decade, GitHub has been a popular way for accessing linked data about code commits and issue/bug information. However, it can be common for large and long-lived open source projects, such as Mozilla and Apache, to rely on their own issue trackers rather than GitHub's, even when the projects provide GitHub mirrors.

In this paper, we presents 20-MAD, a dataset linking the commit and issue data of Mozilla and Apache projects. It includes over 20 years of information about 765 projects, 3.4M commits, 2.3M issues, and 17.3M issue comments. For comparison Ortu et al. [12] dataset has 700K issues and 2M issue comments but no commit information. Commits and issues are linked in two ways: user ids belonging to the same developer[1] in code repositories and issue trackers, and through issue id that is available in over 80% of the commit messages.

In addition to providing linked meta-data about commits and issues, the dataset was processed in various ways:

- Semi-automatic identity merging of developer profiles.
- Timezone information extracted from commit repository and matched with issue tracker timestamps.
- Natural language processing of issue tracker comments including, filtering out of source code with NLoN [10], sentence tokenization with the R package *tokenizers*, emoticon extraction with our own R package[2] and sentiment analysis using SentiStrength [13] and Senti4SD [1].

This dataset originates from our MSR 2017 study on abnormal work hours in Mozilla Firefox [5] for which we extracted Mozilla commit and issue tracker information. We updated it and extended it with Apache code repositories in order to compare work hours in Mozilla projects for an ICSE 2018 publication [6]. We also used it for our MSR 2018 study on paid developers [7]. Meanwhile, we extended the dataset further with issue tracker information in order to perform natural language processing of comments. While we initially considered reusing the Jira dataset from Ortu et al. [12], because of missing accurate timezone information in older Apache commits, we re-extracted Apache's Jira data for our study of emoticons usage by Mozilla and Apache developers [4].

The dataset presented here is an updated version of the dataset we used for our previous studies. It includes open source software development information spanning more than 20 years, going back from January 2020 to 1998 (commits) and 1994 (issues) for Mozilla and 1998 (commits) and 2003 (issues) for Apache. To the best of our knowledge, this is the largest dataset that links commit and issue tracker information without relying on GitHub. In addition, we are not aware of any other dataset providing time zone information in issue trackers. Finally, processing all the issue tracker comments with NLP tools such as NLoN and Senti4SD would take around 20 days of computations with an average laptop.

[1]For data protection, user personal information is anonymized in the dataset.
[2]https://github.com/M3SOulu/EmoticonFindeR

ACM ISBN xxx-x-xxxx-xxxx-x
https://doi.org/10.1145/1234567.89012345

The dataset can be used for various repository mining tasks or studies that would require access to developers weekly and daily work patterns, large scale sentiment analysis, analysis of emoticon usage by software developers or NLP tasks of comments. It can also be used to create a software engineering language model that isn't based on StackOverflow, contrarily to existing ones [9]. While the dataset does not provide any information about source code, it can easily be extended with any other data that keeps track of commit hash in Apache or Mozilla projects.

Our dataset and source code are available in an Open Science Framework repository [2, 3]. The documentation and source code are also available on GitHub[3]. Moreover, we provide a Docker image for simplifying the replication of processing the raw data[4].

In the remaining of this paper, we first present the information available in the dataset in Section 2. Then, in Section 3, we provide details on how the data was extracted and processed. Finally, in Section 4, we conclude by presenting the current limitations of the dataset that researchers should be aware of, and how we plan on improving it in the future.

2 DESCRIPTION OF THE DATASET

The dataset consists of several tables. These are stored as Apache Parquet files[5] in order to keep column type information, such as timestamps. There are two main sets of tables: logs and natural language data. Fig. 1 presents an overview of the dataset with the different Parquet tables and their dependency relationships.

Logs The logs contain meta-data information about commits, issues and issue comments:

Commits Meta-data information about commits. A commit is uniquely identified by its source (i.e., Apache or Mozilla), its repo(sitory) and its hash.

Issues Meta-data information about issues. An issue is uniquely identified by its source, its product tag, and its id. Some fields are only available for issues coming from Jira or Bugzilla.

Comments Meta-data information about issue comments. A comment is uniquely identified by its source (i.e., Apache or Mozilla), its product tag, its issue id and its id.

Timestamps Aggregated timestamp information from the three previous logs. A timestamp is uniquely identified by its source, its repository or product tag, its id[6] and its type and action.

NLP We ran various natural language processing tools on the issue comments:

nlcomments Part of comments that have been detected as natural language. Each comment is split by paragraphs and uniquely identified by its source, product tag, issue id, comment id and paragraph id.

emoticons Emoticons and emojis found in natural language paragraphs. Each emoticon is uniquely identified by its source, product tag, issue id, comment id, paragraph id and its location in the paragraph. The type indicates whether it is a text emoticon or a Unicode emoji.

sentences NL paragraphs are further split as sentences which are uniquely identified by source, product tag, issue id, comment id, paragraph id and sentence id.

sentistrength Result of SentiStrength on each sentence using the default lexicon (min. and max. valence) and our lexicon [11] for detecting arousal in software engineering context (min. and max. arousal).

senti4sd Result of running Senti4SD on each sentence.

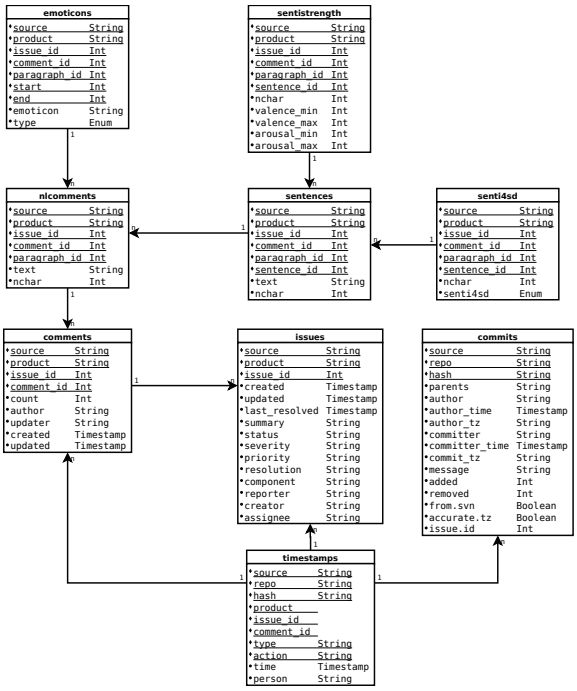

Figure 1: Simplified schema of the main Parquet files of the dataset. More information about the different fields can be found on the GitHub page of the dataset.

3 DATA COLLECTION

In this section, we present how the data was collected and pre-processed. First, how the raw data from Git repositories and issue trackers was extracted using Perceval [8] and the basic pre-processing applied to it. Second, we present the more elaborate processing steps including identity merging, time zone matching and natural language processing (NLP). Fig. 2 presents the workflow followed for data collection as a dependency graph.

3.1 Data extraction

The Git repositories were extracted between January 6th and 9th 2020. Mozilla's Bugzilla data was initially extracted in January 2017 and regularly updated until January 14th 2020. While the same was true for Apache's Jira, because of data corruption in 2019, it was completely re-extracted between January 14th and 20th 2020. Our estimations for processing all the data is around

[3]https://github.com/M3SOulu/MozillaApacheDataset/releases/tag/msr2020
[4]https://hub.docker.com/repository/docker/claesmaelick/mozilla-apache-dataset
[5]https://parquet.apache.org/
[6]Hash for commits, issue id for issues and issue and commit ids for comments.

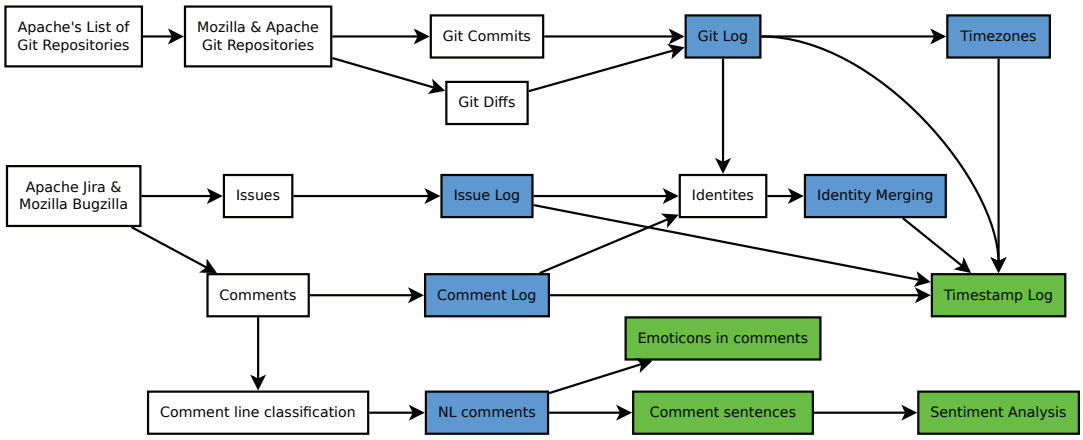

Figure 2: Dependency graph of the different steps for extracting and processing the data. Green rectangles represent the final processed data and blue rectangles pre-processed data that is also provided.

20 days[7]. However, we relied on HPC resources[8]. The 16 days of computations estimated for running Senti4SD were reduced to less than 36 hours using 16 threads and up to 272GB of memory on the HPC environment.

3.1.1 Git commits. We parsed Apache's list of Git repositories[9] using a custom-made Python script. This list provides the name and URL of the 1648[10] Git repositories currently maintained by the Apache community and groups them by projects. For Mozilla, we cloned the GitHub mirrors of the two main Mercurial repositories used by Mozilla: gecko-dev[11] and comm-central[12].

We cloned locally each repository and generated a log file containing all commits[13]. Each repository's generated commit log was parsed as a JSON file using Perceval [8]. To make sure that each individual file stays small, we split the JSON file of Mozilla's gecko-dev into several smaller JSON files of 10,000 commits. Each JSON file was then converted to two Parquet tables: one containing the list of commits of the repository, and one containing the list of files changed in each commit. Time data (Git's commit and author dates) is also parsed as a timestamp and timezone.

3.1.2 Issue trackers. We used Perceval [8] to extract issues from Apache's Jira repository[14] and Mozilla's Bugzilla repository[15] and stored the result in a MongoDB database. We exported the issues from the database into JSON files grouped by the product tag of each issue. Each product's JSON file was then further split into several files of maximum 10,000 issues.

Each JSON file was then converted to three or four Parquet tables: one containing the list of issues and associated meta-data, one containing the issue comments and associated meta-data, one containing the history of changes of the issues, and one containing the versions associated with the issues (only for Jira). The various time data associated with issues and comments (e.g. creation date, update date, resolved date) is also parsed as a timestamp and timezone.

3.2 Data processing

3.2.1 Logs. We produced several logs, lists of records of events, that aggregate the different Parquet tables generated before into a single one:

- A commit log that merges commit log and diff by aggregating the total number of lines added and removed in each commit. In addition, issue tracker ids were extracted from commit messages in order to link commits and issues.
- An issue log that aggregates the issue meta-data from Jira and Bugzilla and fills the missing field with empty values.
- An issue comment log containing meta-data of all issue comments. The text of the comment is removed so metadata of all comments can fit as a single table in memory on a laptop with 8GB memory.

3.2.2 Merging identities. One of the goals of this dataset is to link commit and issue information. In order to make this possible, it is needed to link developer profiles used in code repositories and issue trackers. While issue trackers already have profiles that can be identified for each person, Git repositories only store a character string, usually formatted as *Full Name <emailadress>*. Thus, it is common for a single developer to use multiple distinct identities.

We performed identity merging using a simple semi-automatic technique. We gathered the name and emails of all identities used in issue trackers (issue creators, updaters and assignees, and issue comment authors and updaters). For commits, we split the author and committer fields using regular expressions to obtain separated names and emails. For each identity gathered made of distinct lower-cased names and emails, we create a node in an undirected graph. Then we add edges between each node for which the name

[7] On a laptop with 8GB of memory and an SSD drive.
[8] Provided by CSC (https://wwww.csc.fi)
[9] https://issues.apache.org/jira
[10] The number of Apache repository in the dataset is actually smaller due to empty Git repository listed on the website.
[11] https://github.com/mozilla/gecko-dev
[12] https://github.com/mozilla/releases-comm-central
[13] With the following command: git log –raw –numstat –pretty=fuller –decorate=full –all –reverse –topo-order –parents -M -C -c -l100000 –remotes=origin
[14] https://issues.apache.org/jira
[15] https://bugzilla.mozilla.org/home

or the email is the same. Each connected component of the graph represents a distinct developer profile.

The result was manually verified and edges were manually added and removed to make sure that

- distinct profiles were not merged together;
- the most active developers (in terms of number of commits) were properly linked to a profile in the issue tracker.

This manual verification was conducted for developers with more than 100 commits for the previous version of the data (extracted in January 2018 and used in our ICSE paper [6]), and re-checked for developers with more than 1000 commits for the current version. Overall, it increased the number of distinct (automatically merged) developer profiles by 198 for Mozilla and 258 for Apache.

This technique doesn't provide a perfect identity merging, yet we consider it to be good enough for studying the activity of regular developers in both issue and commit repositories. Table 1 shows the numbers and percentages of developer profiles in Apache and Mozilla that have been linked to a profile in the associated issue tracker. We report a higher percentage for Mozilla than Apache as Mozilla developers have to open an issue in the Bugzilla repository in order to submit any code and get it reviewed. On the other hand, this is not required for Apache developers as individual projects can decide how to handle bug reports and code submissions.

Table 1: Numbers and percentages of developer and issue tracker profiles.

	Mozilla	Apache
# developers without merging	9,894	40,201
# merged developers	6,810	27,627
# issue tracker profiles	271,816	139,434
% developers in issue tracker	68	44.8
% commits from developers in issue tracker	84	83.7
% comments from developers	58.4	55.5

3.2.3 Handling time zones. Because Apache and Mozilla relied on SVN and CVS before Git and Mercurial, not all commits have accurate time zones. For some part of the commit history, time zones are simply missing. First, Apache commits imported from SVN could be identified using regular expressions in the commit messages. Second, for each repository, we only consider time zones being accurate starting from the first commits after 2007 that doesn't use UTC as a timezone. This leaves 78.7% of Mozilla's commit time zones and 59.8% of Apache's commit time zones usable.

Relying on the identity merging of developer profiles, we inferred time zones for the issue tracker timestamps using the ones used in the commits. For each developer, we listed the time zones they used in commits from all repositories from a given source (Mozilla or Apache). Then, for each timestamp in the issue tracker, a time zone is chosen based on the one used by the developer's previous commit. In total, we could infer time zones for 43.4% and 55.4% of all issue and issue comment timestamps for Mozilla, and 57.6% and 50.7% for Apache.

3.2.4 NLP on comments. Each raw comment was processed with different NLP tools:

- Text line classification. We use simple regular expressions to detect lines of text that are automatically generated (only for Bugzilla), quoted or written by the author. In addition, NLoN is used to predict whether each line is natural language or not.
- Natural language filtering. All text that wasn't authored by the comment author (generated or quoted text) and all non natural language text (as detected in the previous step) are removed. Lines of text separated by a single line feed are also grouped by paragraphs for Bugzilla. This is done because individual sentences are often split with line feeds.
- Emoticon detection. Unicode emojis and emoticons contained in natural language are detected using regular expressions relying on our own R package[16].
- Sentence detection. The R package *tokenizers* is used to split each paragraph in sentences.
- Sentiment analysis. Both SentiStrength [13] and Senti4SD [1] are run on each sentence.

4 LIMITATIONS AND FUTURE WORK

This dataset comes with several limitations that one needs to be aware of. First, it is not exhaustive. This is particularly true for Apache as some projects are still not using Jira and Git. In particular, the famous Apache httpd web server[17] is not included as it still uses Bugzilla and SVN. While it would be possible to extend the dataset to include it in the future, it wouldn't be possible to infer any timezone. Second, timestamp information is incomplete as any source code that was originally versioned with SVN and CVS, and later imported in Git and Mercurial, is missing time zone information. Third, the identity merging performed is quite rudimentary. While the result was initially manually checked to avoid linking distinct developers with more than 100 commits for the previous version of the data (i.e., extracted in January 2018 and used in our ICSE paper [6]), it was only re-checked for developers with more than 1000 commits for the updated version due to time constraint.

There are also several limitations regarding the natural language processing of issue comments. First, NLoN's prediction model hasn't been retrained with any Apache data. We are aware of several non natural language Apache comments that are considered as natural language for that reason. The same is true in a lesser extent for Mozilla and this will be improved in the future by adding more manually labelled data to NLoN and re-processing the comments. The description of each issue wasn't included in the NLP pipeline for Apache as it is included as meta-data of the issue, while for Mozilla it is included as the first comment of the issue. In the future, we plan on updating the dataset to include this description field as a comment rather than meta-data.

Finally, as we are still using the dataset, in particular for natural language processing, we will keep on releasing extensions and updates of the dataset in the upcoming years.

ACKNOWLEDGMENTS

The authors have been supported by Academy of Finland grants 298020 and 328058.

[16] https://github.com/M3SOulu/EmoticonFindeR
[17] https://httpd.apache.org/

REFERENCES

[1] Fabio Calefato, Filippo Lanubile, Federico Maiorano, and Nicole Novielli. 2018. Sentiment Polarity Detection for Software Development. *Empirical Softw. Engg.* 23, 3 (June 2018), 1352–1382. https://doi.org/10.1007/s10664-017-9546-9

[2] Maëlick Claes. 2020. 20-MAD: Mozilla Apache Dataset. *Open Science Framework* (2020). https://doi.org/10.17605/OSF.IO/KVXR4

[3] Maëlick Claes. 2020. 20-MAD: R package. *Open Science Framework* (2020). https://doi.org/10.17605/OSF.IO/SR56U

[4] Maëlick Claes, Mika Mäntylä, and Umar Farooq. 2018. On the use of emoticons in open source software development. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM 2018, Oulu, Finland, October 11-12, 2018*. 50:1–50:4. https://doi.org/10.1145/3239235.3267434

[5] Maëlick Claes, Mika Mäntylä, Miikka Kuutila, and Bram Adams. 2017. Abnormal working hours: effect of rapid releases and implications to work content. In *Proceedings of the 14th International Conference on Mining Software Repositories*. IEEE Press, 243–247.

[6] Maëlick Claes, Mika Mäntylä, Miikka Kuutila, and Bram Adams. 2018. Do Programmers Work at Night or During the Weekend?. In *Int'l Conf. Software Engineering*.

[7] Maëlick Claes, Mika Mäntylä, Miikka Kuutila, and Umar Farooq. 2018. Towards automatically identifying paid open source developers. In *Proceedings of the 15th International Conference on Mining Software Repositories*. 437–441.

[8] Santiago Dueñas, Valerio Cosentino, Gregorio Robles, and Jesus M. Gonzalez-Barahona. 2018. Perceval: Software Project Data at Your Will. In *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings* (Gothenburg, Sweden) *(ICSE '18)*. ACM, New York, NY, USA, 1–4. https://doi.org/10.1145/3183440.3183475

[9] Vasiliki Efstathiou, Christos Chatzilenas, and Diomidis Spinellis. 2018. Word embeddings for the software engineering domain. In *Proceedings of the 15th International Conference on Mining Software Repositories*. 38–41.

[10] Mika V Mäntylä, Fabio Calefato, and Maelick Claes. 2018. Natural Language or Not (NLoN)-A Package for Software Engineering Text Analysis Pipeline. *arXiv preprint arXiv:1803.07292* (2018).

[11] Mika V Mäntylä, Nicole Novielli, Filippo Lanubile, Maëlick Claes, and Miikka Kuutila. 2017. Bootstrapping a lexicon for emotional arousal in software engineering. In *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*. IEEE, 198–202.

[12] Marco Ortu, Giuseppe Destefanis, Bram Adams, Alessandro Murgia, Michele Marchesi, and Roberto Tonelli. 2015. The JIRA Repository Dataset: Understanding Social Aspects of Software Development. In *Proceedings of the 11th International Conference on Predictive Models and Data Analytics in Software Engineering* (Beijing, China) *(PROMISE '15)*. ACM, New York, NY, USA, Article 1, 4 pages. https://doi.org/10.1145/2810146.2810147

[13] Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. 2010. Sentiment strength detection in short informal text. *Journal of the American society for information science and technology* 61, 12 (2010), 2544–2558.